The MNIST database

In the exericses of the 3$^{\text{rd}}$ *homework, you are to classify handwritten digits from the MNIST database. An instruction of how you are to load the MNIST datsets is given in the following. Note that this exercise contains no questions, and will not serve as material for grading.*

In the exericses of the 3$^{\text{rd}}$ homework, you are to train different networks into classifying handwritten digits. The datasets that you are to use belong to the MNIST database, which contains two datasets. One dataset is a collection of 60 000 digits written by 250 persons. The other dataset is a collection of 10 000 digits written by another group of 250 persons. The digits in MNIST are grayscale images of 28×28 pixels. To each image belongs a classification, telling which of the ten different digits its writer was instructed to write.

The data that you are to analyze is the data you obtain when executing the MATLAB function contained in `LoadMNIST.m`. The function has the following interface:

```
[xTrain, tTrain, xValid, tValid, xTest, tTest] = LoadMNIST(exerciseNumber);
```

The data in "xTrain", "xValid" and "xTest" encode 50 000, 10 000 and 10 000 written digits respectively. These variables are the input patterns of the training set, the validation set and of the test set. In "tTrain", "tValid" and "tTest", the corresponding target values are contained.

The function "LoadMNIST" takes one in-argument; "exerciseNumber". This is due to that the different exercises of the homework requires the MNIST data in different formats. For exercise 1 and 2 you require one data format, for exercise 3 and 4 you require another. To obtain data with the required format, simply set "exerciseNumber" to the number of the exercise that you are doing.

Exercise 1 and 2

For exercise number 1 and 2, you set "exerciseNumber" to either 1 or 2. "LoadMNIST" then returns "xTrain", "xValid" and "xTest" as arrays of 784 rows and 50 000, 10 000 and 10 000 columns, respectively. Each columns is a 784 element long vector representing a 28×28 pixels image. To show digit number 25081 of the training set, say, you can call the following command:

```
imshow(reshape(xTrain(:,25081),[28 28]));
```

For exercise number 1 and 2, "tTrain", "tValid" and "tTest" are arrays whose elements are either zero or one. Furthermore, they have exactly one element in each column that equals one. If, for the μ^{th} column of a dataset, it is the element at the i^{th} row that equals one, then this encodes that the classification of digit number μ^{th} of this dataset is $i-1$. Thus "tTrain", "tValid" and "tTest" have 10 rows, one for each digit. If you showed digit number 25081 of the training set as explained above, you can call

```
tTrain(:,25081)
```

to check whether or not you agree with its classification.

Exercise 3 and 4:

For exercise number 3 and 4, you set "exerciseNumber" to either 3 or 4. "LoadMNIST" then returns "xTrain", "xValid" and "xTest" in the format suited for MATLAB:s Deep Learning Toolbox . The data sets consist of 28×28×1 ×n 4-dimensional uint8 (8 byte unsigned integer) arrays with n = 50000, 10000 and 10000. The first two dimensions constains 28×28 pixels images with uint8 resolution, i.e. each pixel is represented by an integer from 0 (black) to 255 (white). The arrays are only one layer deep along the 3$^{\text{rd}}$ dimension since we operate with grayscale digits. For color images this dimension spans 3 layers, so that color pixels in RGB format can be represented. The last dimension of an array with input data spans all images in the correspoding set. To show, for instance, image number 25081 you can call the following command:

```
 imshow(mat2gray(xTrain(:,:,1,25081)))
```

For exercise number 3 and 4, "tTrain", "tValid" and "tTest" are categorical 1-dimensional arrays of size 50000, 10000, and 10000. There are in total 10 categories correspoding to the digits from 0 to 9.